AMENDMENTS TO THE CLAIMS

Re-write the claims as set forth below. This listing of claims will replace all prior versions and listings, of claims in the application:

1. (Currently Amended) A method comprising:

hashing an instruction that comprises an operation code and corresponding operand value numbers to generate a first hash value;

retrieving an operation value number from a first hash table based on the first hash value, said first hash value mapping to said operation value number in said first hash table, wherein said operation value number corresponds to an operation of said operation code on said corresponding operand value numbers, and wherein said operation value number further corresponds to components contained by a superword register;

generating a result value number based on a previous value number and the operation value number wherein said result value number is a combination of operation value numbers, wherein said previous value number corresponds to at least one additional component contained by said superword register that was not changed by said operation of said operation code on said corresponding operand value numbers; and

determining if an instruction, wherein said instruction comprises an operation code and corresponding operand value numbers, is redundant by searching a second hash table using the result value number.

2. (previously presented) The method of claim 1 further comprising:

when the result value number is found within the second hash table, retrieving an output of the instruction from the second hash table.

3. (previously presented) The method of claim 1 further comprising:

when the result value number is not found within the second hash table, writing the result value number to the second hash table.

4. (previously presented) The method of claim 1 further comprising:

prior to generating a result value number, retrieving the previous value number.

5. (original) The method of claim 1 further comprising:

prior to retrieving the operation value number, comparing the first hash value with a first hash table.

6. (previously presented) The method of claim 5 further comprising:

when the first hash value is not within the first hash table, assigning the first hash value a multiple component hash value.

7. (Original) The method of claim 1 wherein the operation value number is an n-tuple number.

8. (Currently Amended) ~~The method of claim 1~~ A method comprising:

hashing an operation code and corresponding value numbers to generate a first hash value;

retrieving an operation value number from a first hash table based on the first hash value wherein said operation value number corresponds to components contained by a superword register;

generating a result value number based on a previous value number and the operation value number wherein said result value number is a combination of operation value numbers; and

determining if an instruction is redundant by searching a second hash table using the result value number wherein~~:~~ the instruction further includes a write mask~~;~~ and wherein

generating the result value number includes, for each component in the write mask:

if the write mask value is false, setting the result value number equal to the operation value number; and

if the write mask value is true, setting the result value number equal to the previous value number.

9. (previously presented) The method of claim 1 wherein the instruction further including a previous bit and a write mask.

10. (Currently Amended) An apparatus for superword register value numbering, the apparatus comprising:

at least one memory device storing a plurality of executable instructions wherein each instruction of said plurality of instructions comprises an operation code and corresponding operand value numbers; and

at least one processor operably coupled to the at least one memory device, operative to receive the plurality of executable instructions such that the processor, in response to the executable instructions and for an instruction having an operation code and value numbers of a plurality of sources:

hashes an operation code and corresponding operand value numbers to generate a first hash value;

retrieves an operation value number from a first hash table based on the first hash value, said first hash value mapping to said operation value number in said first hash table, wherein said operation value number corresponds to an operation of said operation code on said corresponding operand value numbers, and wherein said operation value number further corresponds to components contained by a superword register;

generates a result value number based on a previous value number and the operation value number wherein said result value number is a combination of operation value numbers, wherein said previous value number corresponds to at least one additional component contained by said superword register that was not changed by said operation of said operation code on said corresponding operand value numbers; and

determines if the instruction is redundant by searching a second hash table using the result value number.

11. (previously presented) The apparatus of claim 10 wherein the at least one processor further in response to the executable instructions:

when the result value number is found within the second hash table, retrieves an output of the instruction from the second hash table.

12. (previously presented) The apparatus of claim 10 wherein the at least one processor further in response to the executable instructions:

when the result value number is not found within the second hash table, writes the result value number to the second hash table.

13. (previously presented) The apparatus of claim 10 wherein the at least one processor further in response to the executable instructions:

prior to generating a result value number, retrieves the previous value number.

14. (original) The apparatus of claim 10 wherein the at least one processor further in response to the executable instructions:

prior to retrieving the operation value number, compares the first hash value with a first hash table.

15. (previously presented) The apparatus of claim 14 wherein the at least one processor further in response to the executable instructions:

when the first hash value is not within the first hash table, assigns the first hash value a multiple component hash value.

16. (Currently Amended) ~~The apparatus of claim 10~~ An apparatus for superword register value numbering, the apparatus comprising:

at least one memory device storing a plurality of executable instructions; and

at least one processor operably coupled to the at least one memory device, operative to receive the plurality of executable instructions such that the processor, in response to the executable instructions and for an instruction having an operation code and value numbers of a plurality of sources:

hashes an operation code and corresponding value numbers to generate a first hash value;

retrieves an operation value number from a first hash table based on the first hash value wherein said operation value number corresponds to components contained by a superword register;

generates a result value number based on a previous value number and the operation value number wherein said result value number is a combination of operation value numbers; and

determines if the instruction is redundant by searching a second hash table using the result value number wherein~~:~~ the instruction further includes a write mask; and wherein when the at least one processor generates the result value number, the at least one processor further in response to the executable, and for each component in a write mask:

sets the result value number equal to the operation value number if the write mask value is false; and

sets the result value number equal to the previous value number if the write mask value is true.

17. (original) The apparatus of claim 10 further comprising:

a superword register operably coupled to the processor, the superword register operative to store a plurality of instructions therein.

18. (original) The apparatus of claim 10 further comprising:

at least one hash memory device operably coupled to the at least one processor such that the at least one hash memory device is operative to store the first hash table and the second hash table.

19. (Currently Amended) A method comprising:

hashing an instruction that comprises an operation code and corresponding operand value numbers to generate a first hash value;

comparing the first hash value with a first hash table to determine if said first hash value is present in said first hash table;

retrieving an operation value number from the first hash table based on said first hash value, said first hash value mapping to said operation value number is said first hash table, wherein said operation value number corresponds to an operation of said operation code on said corresponding operand value numbers, and wherein said operation value number further corresponds to components contained by a superword register;

retrieving a previous value number, wherein said previous value number corresponds to at least one additional component contained by said superword register that was not changed by said operation of said operation code on said corresponding operand value numbers;

generating a result value number based on the previous value number and the operation value number wherein said result value number is a combination of operation value numbers;

searching a second hash table using the result value number;

if the result value number is found within the second hash table, retrieving an output of an instruction from the second hash table; and

if the result value number is not found within the second hash table, writing the result value number to the second hash table.

20. (previously presented) The method of claim 19 further comprising:

when the first hash value is not within the first hash table, assigning the first hash value a multiple component hash value.

21. (Currently Amended) ~~The method of claim 19~~ A method comprising:

hashing an operation code and corresponding value numbers to generate a first hash value;

comparing the first hash value with a first hash table;

retrieving an operation value number from the first hash table;

retrieving a previous value number;

generating a result value number based on the previous value number and the operation value number wherein said result value number is a combination of operation value numbers;

searching a second hash table using the result value number;

if the result value number is found within the second hash table, retrieving an output of an instruction from the second hash table; and

if the result value number is not found within the second hash table, writing the result value number to the second hash table; wherein~~:~~ the instruction further includes a write mask; and wherein generating the result value number includes, for each component in the write mask:

if the write mask value is false, setting the result value number equal to the operation value number; and

if the write mask value is true, setting the result value number equal to the previous value number.